

RECEIVED
CENTRAL FAX CENTER
SEP 13 2006

Amendments to the claims,

Listing of all claims pursuant to 37 CFR 1.121(e)

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently amended) A method for executing a database statement, the method comprising:

preparing at least one template for capable of generating an executable statement for execution of a statement against a database via a particular database connection;

storing said at least one template in a shared cache at an application server so that it is available to a plurality of database connections, such that memory requirements of said shared cache are proportional to how many templates exist, regardless of how many executable statements are generated;

in response to a request to execute a particular statement on a given database connection, determining whether a template for said particular statement is available in the shared cache;

if the template is available in the shared cache, creating a corresponding executable database statement based on the template, the executable statement having been prepared for execution on the given database connection; and

executing the database said corresponding executable statement on the given database connection.

2. (Original) The method of claim 1, wherein said preparing step comprises preparing a structure for execution of the statement against a database.

3. (Original) The method of claim 2, wherein said structure comprises an executable structure for executing the statement on a database connection.

4. (Currently amended) The method of claim 1, wherein the request to execute a particular statement on a given database connection is received at an application server.

5. (Original) The method of claim 1, wherein the request to execute a particular statement on a given database connection is received at a database client.

6. (Currently amended) The method of claim 1, wherein said shared cache is available at a database client to application threads via a particular database connection at the application server to application threads accessing a remote database server.

7. (Cancelled)

8. (Original) The method of claim 1, wherein said given database connection includes a connection allocated from a connection pool.

9. (Original) The method of claim 1, further comprising:
if the template is unavailable, preparing a template for execution of the particular statement against the database;
placing the template in the shared cache; and
creating a database statement based on the template for execution on the given connection.

10. (Original) The method of claim 1, wherein said storing step includes making said shared cache available on a first database connection.

11. (Original) The method of claim 10, wherein a template in said shared cache is cloned for use on a second database connection.

12. (Original) The method of claim 1, further comprising:
assigning a unique identifier to each statement to be executed against the database.

13. (Original) The method of claim 12, wherein said determining step includes using said unique identifier to determine whether a template is available in the shared

cache.

14. (Original) The method of claim 12, wherein said shared cache comprises an array of templates indexed based on said unique identifier.

15. (Original) The method of claim 1, further comprising:
synchronizing access to the shared cache from application threads on a plurality of database connections.

16. (Original) The method of claim 1, wherein said creating step includes reusing immutable portions of a template.

17. (Original) The method of claim 1, wherein said creating step includes duplicating mutable portions of a template.

18. (Original) The method of claim 17, wherein said mutable portions include nodes of the template having mutable children.

19. (Original) The method of claim 17, wherein said mutable portions include nodes of the template having mutable fields.

20. (Original) The method of claim 1, wherein said creating step includes attaching the database statement to the given database connection.

21. (Original) The method of claim 1, wherein said executing step includes returning results of executing the database statement.

22. (Original) A computer-readable medium having processor-executable instructions for performing the method of claim 1.

23. (Currently amended) The method of claim 1, further comprising:

downloading a A downloadable set of computer-executable instructions for performing the method of claim 1.

24. (Currently amended) A system for executing a database statement, the system comprising:

at least one template comprising an executable structure for generating an executable statement for execution of a statement against a database;

a shared cache located in a middle tier for storing said at least one template attached to a first connection, such that memory requirements of said shared cache are proportional to how many templates exist, regardless of how many executable statements are generated;

a locator module for locating a template corresponding to a particular statement in the shared cache in response to a request to execute the particular statement on a second database connection;

a cloning module for cloning the template to create a database for creating an executable database statement for execution on the second database connection; and

an execution module for executing the executable database statement on the second database connection.

25. (Original) The system of claim 24, wherein said at least one template comprises at least one database prepared statement.

26. (Original) The system of claim 24, wherein said at least one template comprises at least one prepared statement in executable form for execution against a database.

27. (Original) The system of claim 24, wherein the request to execute a particular statement on a second database connection is received at an application server.

28. (Original) The system of claim 24, wherein the request to execute a particular statement on a second database connection is received at a database client.

29. (Currently amended) The system of claim 24, wherein said shared cache is available ~~at a database client to application threads via a particular database connection at an application server to application threads accessing a remote database server.~~

30. (Canceled)

31. (Original) The system of claim 24, wherein said second database connection includes a connection allocated from a connection pool.

32. (Original) The system of claim 24, further comprising:
a module for preparing a template for execution of the particular statement and placing the template in the shared cache if the locator module determines that the template is unavailable.

33. (Original) The system of claim 24, further comprising:
a module for assigning a unique identifier to a statement to be executed against the database.

34. (Original) The system of claim 33, wherein the locator module uses said unique identifier to determine whether a template is available in the shared cache.

35. (Original) The system of claim 33, wherein said shared cache comprises an array of templates indexed based on said unique identifier.

36. (Original) The system of claim 24, further comprising:
a module for synchronizing access to the shared cache from application threads on a plurality of database connections.

37. (Original) The system of claim 24, wherein the cloning module reuses immutable portions of a template in cloning the template.

38. (Original) The system of claim 24, wherein the cloning module duplicates mutable portions of a template in cloning the template.

39. (Original) The system of claim 38, wherein said mutable portions of a template include nodes of the template having mutable children.

40. (Original) The system of claim 38, wherein said mutable portions of a template include nodes of the template having mutable fields.

41. (Original) The system of claim 24, wherein the cloning module attaches the databasc statement to the second databasc connection.

42. (Original) The system of claim 24, wherein the execution module returns any results of executing the database statement.

43. - 62. (Cancelled)